

***SCHOOL OF ELECTRONIC ENGINEERING***

电 子 工 程 学 院

**《图 像 分 析 与 处 理》**

**实验指导**

西安邮电大学

电 子 工 程 学 院

2025年4月

**实验二**  **基于MWORKS.Syslab的数字图像频域滤波**

一、实验目的与要求

1．熟悉及掌握图像滤波的基本定义及目的。

2．理解傅里叶变换的原理，掌握傅里叶变换的性质。

3. 掌握用MWORKS.Syslab完成数字图像傅立叶变换及逆变换的方法。

4. 掌握频域平滑原理，会用MWORKS.Syslab实现对数字图像的理想低通滤波和

Butterworth低通滤波。

5. 掌握频域锐化原理，会用MWORKS.Syslab实现对数字图像的理想高通滤波和

Butterworth高通滤波。

二、实验原理及知识点

MWORKS正在成为中国的MATLAB，是全球第四个科学计算与建模仿真一体化软件，其中MWORKS.Syslab用于算法开发与数值计算、数据分析与可视化、信息域计算分析等场景，提供基础数学、符号数学、统计、优化等大量高质量内置函数，支撑科学与工程计算，使用的算法语言是Julia语言，Julia 语言是一门高级、通用、动态的程序设计语言，适合用于与科学计算相关的高性能计算领域。它以易于应用的环境集成了计算、可视化和编程。作为强大的科学计算平台，它几乎能够满足所有的计算需求。

MWORKS.Syslab信号处理工具箱旨在为信号在时、频域处理与分析提供各种常用算法和应用程序，包括信号预处理、特征提取和测量、滤波器设计和分析和频谱分析等多项功能。其中信号预处理能够完成对信号去噪、平滑和去趋势处理，为进一步分析做好准备；特征提取和测量能够完成测量和提取信号中的独特特征，包括峰值、功率、带宽、失真和信号统计信息，同时支持计算与脉冲和瞬态相关的指标；滤波器设计和分析提供设计、分析和实现数字和模拟滤波器，支持使用滤波器设计应用程序或函数来设计各种数字 FIR 和 IIR 滤波器，如低通、高通和带阻滤波器；频谱分析支持使用频谱估计和子空间方法表征信号的频率成分，以及设计和分析数据窗。

1. 傅里叶变换 (一维、二维、多维)

使用函数库TyMath中的fft函数完成信号的傅里叶变换，fft的语法为

Y = fft(X)

用快速傅里叶变换 (FFT) 算法计算X的离散傅里叶变换 (DFT)：如果X是向量，则返回该向量的傅里叶变换；如果X是矩阵，则返回该矩阵的二维傅里叶变换；如果X是一个多维数组，返回该多维数组的N维傅里叶变换。二维傅里叶变换对处理二维信号和其他二维数据（如图像）很有用。

1. 傅里叶逆变换 (一维、二维、多维)

使用函数库TyMath中的ifft函数完成信号的傅里叶逆变换，ifft的语法为：

X = ifft(Y)

使用快速傅里叶变换算法计算Y的逆离散傅里叶变换。X与Y的大小相同。如果Y是向量，则返回该向量的逆变换；如果Y是矩阵，则返回对矩阵的二维逆变换；如果Y是多维数组，则返回对多维数组的N维逆变换。

1. 将零频分量移到频谱中心
2. 使用函数库TyMath中的fftshift函数将零频分量移到频谱中心，fftshift的语法为：

Y = fftshift(X)

通过将零频分量移动到数组中心，重新排列傅里叶变换X。如果X是向量，则fftshift会将X的两半部分进行交换；如果X是矩阵，则fftshift会将X的第一象限与第三象限交换，将第二象限与第四象限交换；如果X是一个多维数组，则fftshift会沿每个维度交换X的半空间。

1. 使用函数库TyMath中的ifftshift函数将零频分量移到频谱中心，ifftshift的语法为：

X = ifftshift(Y)

将进行过零频平移的傅里叶变换 Y 重新排列回原始变换输出的样子。换言之，ifftshift 就是撤消fftshift的结果。如果Y是向量，则ifftshift会将Y的两半部分进行交换；如果Y是矩阵，则ifftshift会将Y的第一象限与第三象限交换，将第二象限与第四象限交换；如果 Y是多维数组，则ifftshift会沿每个维度交换Y的半空间。

1. 数字滤波器的频率响应

使用函数库TySignalProcessing中的freqz函数求得数字滤波器的频率响应，freqz的语法为：

freqz(\_\_\_; plotfig = true)

在没有输出参数的情况下，绘制滤波器的频率响应。

示例：设计一个通带（阻带）为 0.35π≤ω≤0.65π rad/sample 的 48 阶 FIR 带通（带阻）滤波器。可视化其幅度和相位响应。

b = fir1(48, [0.35 0.65],"bandpass")

freqz(b, [1],512,plotfig = true)

b = fir1(48, [0.35 0.65],"bandstop")

freqz(b, [1],512,plotfig = true)

函数fir1定义：

fir1(n,Wn,ftype,window)

ftype：”lowpass ” | ”highpass” | ”bandpass” | ”bandstop”

window：指定窗，默认汉宁窗

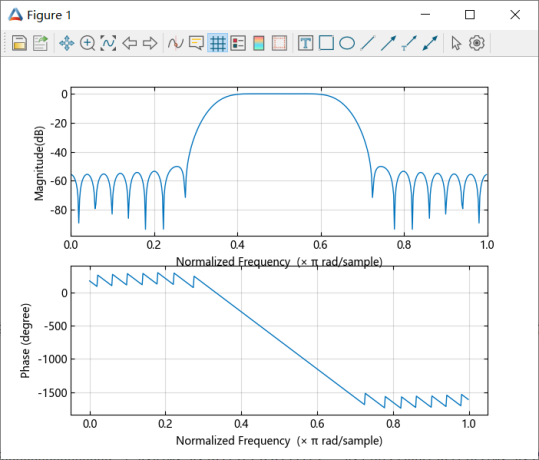
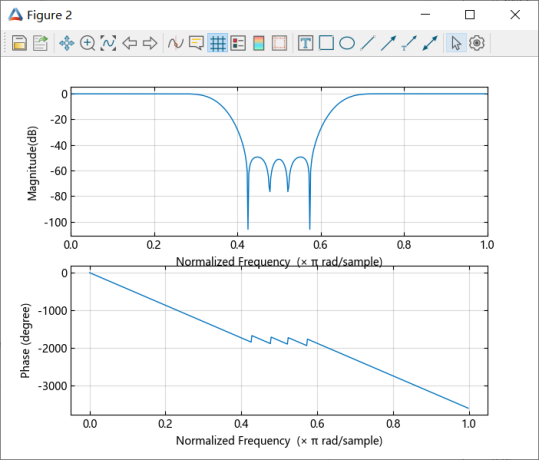
 

图1 48阶FIR带通滤波器幅度和相位响应 图2 48阶FIR带阻滤波器幅度和相位响应

1. 创建曲面图
2. 使用函数库TyPlot中的surf函数创建曲面图，surf的语法为：

surf([Z](file:///C:/Program%20Files/MWORKS/Syslab%202024b/Docs/Help/TyPlot/index.html" \l "/Doc/TyPlot/2Dand3DPlots/SurfacesVolumesandPolygons/surf.html#%E8%AF%AD%E6%B3%95#/Doc/TyPlot/2Dand3DPlots/SurfacesVolumesandPolygons/surf.html#z-z坐标))

创建一个曲面图，并将Z中元素的列索引和行索引用作x坐标和y坐标。

1. 使用函数库TyBase中的meshgrid2函数创建二维网格，meshgrid2的语法为：

X,Y =meshgrid2(x,y)

基于向量x和y中包含的坐标返回二维网格坐标。X是一个矩阵，每一行是x的一个副本；Y也是一个矩阵，每一列是y的一个副本。坐标X和Y表示的网格有length(y)个行和 length(x)个列。

示例：创建三个相同大小的矩阵。然后将它们绘制为一个曲面。曲面图对高度和颜色均使用Z。

using TyBase

using TyPlot

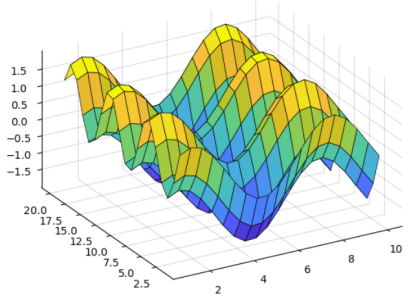
X, Y = meshgrid2(1:0.5:10, 1:20)

Z = sin.(X) + cos.(Y)

fig = figure( )

ax = subplot(; projection="3d")

surf(ax, X, Y, Z)

 图3 示例曲面图

1. 创建网格曲面图

使用函数库TyPlot中的mesh函数创建曲面图，mesh的语法为：

mesh(Z)

创建一个网格图，并将Z中元素的列索引和行索引用作x坐标和y坐标。

示例：创建三个相同大小的矩阵。然后将它们绘制为一个网格图。该绘图使用Z确定高度和颜色。

using TyBase

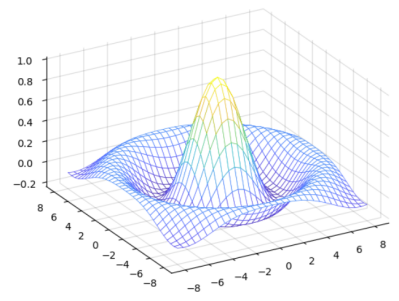
using TyPlot

X, Y = meshgrid2(-8:0.5:8, -8:0.5:8)

R = sqrt.(X .^ 2 .+ Y .^ 2) .+ eps()

Z = sin.(R) ./ R

mesh(X, Y, Z)

 图4 示例网格曲面图

1. 傅里叶变换性质

图像经过傅里叶变换后，得到的是图像的频域，也就是频率成分；这个频率成分表示的意义就是相邻像素之间数值的变化，也就是说像素在空间上的变化越快，它对应在频域上的数值就越大；如果频谱图中亮的点数多，那么实际图像一定是尖锐的，边界分明且边界两边像素差异较大的。对频谱移频到原点以后，可以看出图像的频率分布是以原点为圆心，对称分布的。

频域图像滤波的步骤：首先将图像变换到频域，然后在频域进行需要的滤波，即和滤波器转移函数相乘，最后反变换回时域得到增强的图像。常用的频域增强方法有低通滤波和高通滤波。

1. 低通滤波

一般来说，图像的边缘和噪声都对应于傅立叶变换中的高频部分，所以能够让低频信号畅通无阻而同时滤掉高频分量的低通滤波器能够平滑图像，去除噪声。常用的几种有，理想的低通滤波器，Butterworth低通滤波器，高斯低通滤波器等。传递函数形式如下所示。

理想的低通滤波器：

Butterworth低通滤波器：



表示到原点的距离，表示截止频率点到原点的距离。傅立叶变换的主要能量集中在频谱的中心，合理的选择截止频率对保留图象的能量至关重要。理想的低通滤波后的图象将会出现一种“振铃”特性，造成图象不同程度的模糊，越小，模糊的程度越明显。造成这种模糊的原因在于理想的低通滤波器的传递函数在处由1突变为0，经傅立叶反变换后在空域中表现为同心圆的形式。

1. 高通滤波

图像中灰度发生骤变的部分与其频谱的高频分量相对应，所以采用高通滤波器衰减或抑制低频分量，高频分量畅通并对图像进行锐化处理。常用的高通滤波器有理想的高通滤波器、Butterworth高滤波器，高斯高通滤波等。传递函数如下。

理想的高通滤波器：

Butterworth高通滤波器：



由于经过高通滤波后图像丢失了许多低频信息，所以图像平滑区基本会消失。为此，需要采用高频加强滤波来弥补。高频加强滤波是在设计滤波传递函数时在原有设计结果上条件一个大于0小于的常数即：



滤波结果为：



可见高频加强滤波在高通滤波的基础上保留了的低通分量，效果比一般的高通滤波要好。

三、实验内容及步骤

1．利用imread( )函数读取一幅图像，将图像数据存入一个数组中；

2．利用imshow( )函数来显示图像；

3．利用fft( )函数来完成图像信号的傅里叶变换；

4．利用fftshift( )函数将零频分量移到频谱中心；

5．利用mesh( )函数来创建图像的幅度谱图；

6．构建理想低通滤波器，并实现图像的理想低通平滑滤波；

7．构建Butterworth低通滤波器，并实现图像的Butterworth低通平滑滤波；

8．构建理想高通滤波器，并实现图像的理想高通锐化滤波；

9．构建Butterworth高通滤波器，并实现图像的Butterworth高通锐化滤波；

10. 利用ifftshift( )函数进行图像频谱移中的反向操作；

11. 利用ifft( )函数来完成图像信号的傅里叶逆变换；

12. 利用imshow( )函数显示滤波后的图像；

13. 利用mesh( )函数来创建滤波后图像的幅度谱曲面图；

14. 利用surf( )函数来创建不同滤波器的幅度谱曲面图；

15．将以上每一步的函数执行语句拷贝下来，写入实验报告，将得到的对应图像结果拷贝下来放入实验报告，并进行结果分析。

四、考核要点

1. 熟悉在MWORKS.Syslab中如何利用相关函数读入图像并求取图像的傅里叶变换，完成频谱移中并显示对应的幅度谱。

2. 熟悉在MWORKS.Syslab中利用函数如何求取图像的傅里叶逆变换。

3. 熟悉在MWORKS.Syslab中如何实现图像的频域低通及高通滤波处理，显示滤波后的图像及对应的幅度谱，同时显示滤波器的幅度谱，对比分析滤波结果。

五、实验报告要求

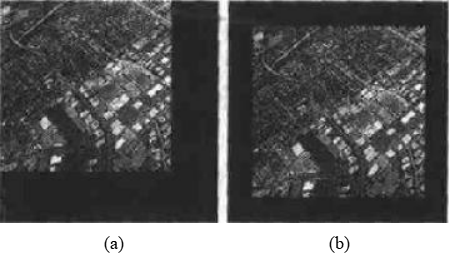
描述实验目的、实验原理及实验基本步骤，用数据和图像给出各个步骤中取得的实验结果，并进行相应的数据分析，就实验中遇到的问题及解决方法进行描述，回答思考题，给出实验总结及程序清单。

六、思考题

(1) 分析为什么图像通过低通滤波后变得模糊？通过高通滤波后得到锐化结果？

(2) 分析不同低通及高通滤波器的滤波效果，说明滤波器的特点。

(3) 频率域滤波时需要图像延拓，需要延拓的图像在图像中行和列的末尾要填充0值，见图(a)。你认为如果把图像放在中心，四周填充0值（见图(b)）而不改变0值的总数，会有区别么？请解释？



七、实验图像

Fig.1 lena.jpg Fig.2 house.jpg



Fig.3 lena\_noise.jpg